Prediction of Student Performance in Abacus-Based Calculation Using Matrix Factorization

Keita Tokuda*
Department of Pharmacy, the
University of Tokyo Hospital, Faculty
of Medicine, the University of Tokyo
Tokyo, Japan
tokudakeita-tky@umin.ac.jp

David Kaschub
Digika Co., Ltd.
Tokyo, Japan
david@digika.jp

Takuma Ota
Digika Co., Ltd.
Tokyo, Japan
ota@digika.jp

Yasunobu Hashimoto
Digika Co., Ltd.
Tokyo, Japan
yas@digika.jp

Naoya Fujiwara
Graduate School of Information
Sciences, Tohoku University
Sendai, Japan
fujiwara@se.is.tohoku.ac.jp

Akihito Sudo
Faculty of Informatics, Shizuoka
University
Shizuoka, Japan
sudo@inf.shizuoka.ac.jp

ABSTRACT

We conducted modeling of student learning status and tasks in abacus-based calculation by utilizing matrix factorization on student-generated learning data. The matrix consisted of performance scores on student-task pairs. We decomposed the raw matrix into two matrices, yielding the distributed representations of each student and each task. Prediction of student performance using those decomposed matrices achieved better results than baseline models that use the student biases and task biases. This suggests matrix factorization successfully extracted the interaction of multiple latent features of each task and each student's learning status in abacus-based calculation.

CCS CONCEPTS

• HCI design and evaluation methods → User models.

KEYWORDS

performance prediction in education, data accumulation in education, abacus, evaluation methodologies, STEM education

ACM Reference Format:

Keita Tokuda, David Kaschub, Takuma Ota, Yasunobu Hashimoto, Naoya Fujiwara, and Akihito Sudo. 2020. Prediction of Student Performance in Abacus-Based Calculation Using Matrix Factorization. In *Adjunct Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization (UMAP '20 Adjunct)*, July 14–17, 2020, Genoa, Italy. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3386392.3399309>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

UMAP '20 Adjunct, July 14–17, 2020, Genoa, Italy

© 2020 Association for Computing Machinery.

ACM ISBN 978-1-4503-7950-2/20/07...\$15.00

<https://doi.org/10.1145/3386392.3399309>

1 INTRODUCTION

The abacus is a tool for rapid and precise calculation that has been used for thousands of years in East Asia [13]. The Japanese abacus, or “Soroban,” is made of multiple vertical columns, and the positions of the freely-moving beads attached to the columns represent integers [9]. Comprehension of the notion of numbers through this physical representation has been highly regarded as an important skill within the Japanese education community, and acquisition of the basic usage of the abacus is included in the curriculum of compulsory elementary schools in Japan [14]. Additionally, several studies have reported that training on an abacus has positive effects on mathematical skills [2, 19, 20]. Furthermore, by training on an abacus, children in countries such as Japan, China, India, and Singapore often acquire the ability to execute very complex mental arithmetic computations with remarkable speed and accuracy, and this ability is referred to as “mental abacus” [5, 9, 15]. Because of the remarkable cognitive skills realized through mental abacus, it has attracted great interest in various fields, such as cognitive science [1, 5–7] and neuroscience [4, 8].

Modeling the learning status of individual students and individual tasks would be highly beneficial in education. For example, successful modeling of students and tasks will lead to the better prediction of student performance for a specific task that the student has yet to perform. That would in turn allow for the optimization of the difficulty of the tasks to give to each student. Another possible application is to characterize each student's strengths and weaknesses based on the model: this would let each student focus on the specific points necessary for him or her. Thus, the benefits of successfully modeling students can be capitalized on by using adaptive learning systems and lead to the deeper understanding of the cognitive process involved in learning the subject.

The most typical kind of data acquired in educational systems are student performance records, such as the number of correct responses on a particular assignment. Thus, estimating the learning status of each student based on the performance data in his or her learning history is desirable. However, the status of the student is not directly expressed in the data. Rather, the performance score in the data is always given to a specific combination of a student

and a task. Thus, one has to dissociate the effect of student status and the effect of contents of the task on each performance in a student-task pair. Another problem in user modeling employing the learning history data is that a single task or an individual student should be described with latent multiple components, rather than a single scalar value. For example, a single arithmetic task such as $4 + 1 + 3 - 5$ can be described by multiple features such as the number of additions, subtractions, and carry-ups. A student may be strong in addition and weak in subtraction. However, we do not know the proper features to characterize students and tasks *a priori*. This problem is shared with the user preference prediction problem within fields other than education. The difficulty in matching features of the model to the features of the user-content pair is a crucial problem for industries that need effective recommendation systems, such as movies, products, and travel. To overcome these problems, Nguyen et al. pointed out this similarity and introduced matrix factorization in user modeling in education, which has been successful in modeling users and content in recommendation systems such as movies [17]. Their report showed that the matrix factorization technique yields better accuracy in performance prediction than conventional modeling in education, such as the logistic regression model.

Because of the remarkable skill of the abacus experts in calculation, many studies have elucidated the cognitive process of the abacus-based calculation. However, to our knowledge, there have not been any studies trying to model the learning status of individual learners of the abacus with the data-driven approach, presumably due to the lack of data. Digika Co., Ltd., an educational technology company in Japan, has developed an educational system for learning abacus-based mental calculation called “SoroTouch,” which is implemented as an iPad application [18]. SoroTouch enabled the accumulation of vast amounts of learning data from a large number of students every day. Data are recorded at the level of individual students and individual calculations. Their rich data offer the chance to conduct a data-driven approach to student modeling in the learning of abacus-based calculation. Successful user modeling in the acquisition process of the abacus-based calculation should contribute to a better understanding of the cognitive process of calculation in humans and the realization of effective educational strategies for developing computational skills in children.

In this study, matrix factorization was applied to the performance data for the student-problem combination obtained from students using SoroTouch, and a model for predicting the performance of the unknown student-task combination was constructed.

2 PROBLEM SETTING

In the current study, our problem setting is to predict the performance score of a specific student in a specific task. As the performance score, we use the percentage of correct answers the student gives. The matrix shown at the upper left in Figure 1 illustrates the input data of the model. The (s, t) th entity of the matrix represents the percentage of correct answers the s th student gave to the t th task. The matrix is usually sparse; there are missing values. The specific problem setting here is to predict the value of missing entity in the input matrix.

3 MODEL

Generally, matrix factorization is the task that gives an approximation of a matrix P by a product of two smaller matrices W and H , namely, $P \sim WH^T$, where H^T is the transpose matrix of H [3, 11, 17]. Here, p_{st} , the (s, t) th entry of the matrix P is the performance score of the s th student in the t th task. $W \in \mathbb{R}^{S \times K}$ is the matrix whose s th row represents the latent factor describing the s th student, and $H \in \mathbb{R}^{T \times K}$ is the matrix whose t th row represents the latent factor describing the t th task. In this study, we incorporate additional bias terms for the prediction of the performance as follows:

$$\hat{p}_{st} = \alpha + \beta_s + \gamma_t + \langle \vec{w}_s, \vec{h}_t \rangle, \quad (1)$$

$$= \alpha + \beta_s + \gamma_t + \sum_{k=1}^K w_{sk} h_{tk}, \quad (2)$$

where $\hat{p}_{st}$ is the prediction of the performance of the s th student in the t th task, α is the global average of the performance, β_s is the bias term for the s th student, γ_t is the bias term for the t th task, $\vec{w}_s$ and $\vec{h}_t$ are the K -dimensional latent factor vector representing the learning status of the s th student and the t th task, respectively. The model is illustrated in Figure 1. Incorporation of the bias terms enables direct assessment whether the matrix factorization extracts the interaction of student and task, by comparison of the model performance with the baseline models without the factorization term (Section 4.2). K is the dimension of the latent feature vector.

4 EXPERIMENT

4.1 Dataset

4.1.1 iPad application for Abacus-based calculation. We used the learning history data acquired by SoroTouch, the educational iPad application for abacus-based calculation developed and implemented in education by Digika Co., Ltd. The data were acquired from August 29 2016 to October 10 2019 from 5776 total students and 19269 tasks. All the students are given unique IDs, and whether each student succeeded in solving a problem of arithmetic calculation is recorded in the data with the student ID. The interface of SoroTouch for solving a single task is illustrated in Figure 2. A task is defined in this study by a sequence of arithmetic computations consist of addition, subtraction, multiplication, and division, such as $5+10+3$. In this study, the difference in the order of calculation is not ignored, e.g., $5+10+3$ and $10+5+3$ are treated as different tasks. In this case, students were asked to give the correct answer, 18, by inputting on the touchscreen of the iPad. Because each student was often given a same task at different times or days, there are records of multiple trials for a specific student-task pair. Thus, it is possible to calculate the performance score for a student-task pair by the percentage of the correct answers. The performance score p_{st} %, was calculated from the data as follows:

$$p_{st} = \frac{n_{st}^{\text{correct}}}{n_{st}^{\text{total}}} \times 100 \quad (3)$$

where n_{st}^{correct} and n_{st}^{total} are the number of trials when the s th student gave a correct answer to the t th task, and the number of total trials s th student tried to solve the t th task.

Figure 1: Data structure and the matrix factorization model with the bias terms. The (s, t) th entity of the matrix P , p_{st} , holds the value of performance score calculated from the learning history data. There are missing values in the input matrix (colored in light blue).

Figure 2: SoroTouch interface for a single calculation task. The appearance of the interface when solving a calculation task, $12+26=38$, is shown. The orange bead (shown in panel d) represents a value of 5 and the yellow bead represents a value of 1. The summation over the lighting beads expresses a certain number value. The beads light up by touching the screen. In the case of (a), the two thumbs are touching 12, and (b) shows the touched beads lighting up as the fingers are released. Consequently, (c) shows 26 added on to the initial value of 12, and (d) shows the lighting beads corresponding to the value of 38.

4.1.2 Data pre-processing. Obviously, the value p_{st} is unreliable when the total number of trials the student solved this task, n_{st}^{total} , is small. As the pre-processing, for each student-task pair, data were discarded if n_{st}^{total} was smaller than 40. We also discarded any student who solved less than 10 tasks and any task that is solved by less than 10 students. The resulting size of the valid data was 2442 total students and 626 tasks with performance scores of 115223 student-task pairs (table 1). Note that the matrix is sparse (7.54 %).

Table 1: The number of students, tasks, and performances after pre-processing

# of Students	# of Tasks	# of Performances
2442	626	115223

4.2 Baseline methods

For evaluation of the model performance, the prediction accuracy is compared to the two baseline models [11, 16].

4.2.1 Global average (GA). For global average model (GA), only the bias term in equation (1), μ , is used as follows:

$$\hat{p}_{st} = \mu. \quad (4)$$

In this study, $\mu = 88.3657\%$ was used, which is the mean value of the training data.

4.2.2 Biased student task model (BUI). In biased student task model (BUI), only the bias terms in equation (1) are used as follows:

$$\hat{p}_{st} = \mu + \beta_s + \gamma_t. \quad (5)$$

4.3 Training and validation

The data are divided into a training set and a validation set by randomly choosing the 95 % of all the student-task pairs as the training set (109461 student-task pairs), and the rest are used as the validation set (5762 student-task pairs). The training set comprises the input data with the format shown in the lhs of Figure 1. In the training phase of the model, the parameters α , β_s , γ_t , $\vec{w}_s$, $\vec{h}_t$ are obtained by stochastic gradient descent minimizing the following objective function:

$$O \equiv \sum_{\{s,t\} \in D} (p_{st} - \hat{p}_{st})^2 + \beta(\|W\|_F^2 + \|H\|_F^2), \quad (6)$$

where $D = \{\{s_1, t_1\}, \{s_2, t_2\}, \dots, \{s_N, t_N\}\}$ is the set of indices of the matrix P where the score exists in the training set, $\|\cdot\|_F^2$ is the

Frobenius norm of a matrix, β is the regularization parameter. The values $K = 150$, $\beta = 10^{-3}$ are used in the current study.

4.4 Evaluation metric

The root mean squared error (RMSE) defined below is used as the evaluation metric in this study.

$$\text{RMSE} \equiv \sqrt{\frac{\sum_{\{s,t\} \in D^{\text{test}}} (p_{st} - \hat{p}_{st})^2}{|D^{\text{test}}|}} \quad (7)$$

5 RESULTS

Figure 3 shows the scatter-gram showing the performance predictions for student-task pairs in the validation set plotted against the real performance scores. There is a clear correlation between the prediction and the real values. The Pearson's correlation coefficient was $r = 0.773$. Figure 4 shows the root mean squared error (RMSE) for the two baseline models (GA and BUI) and the matrix factorization model (MF). The RMSE values for BUI and MF were 68.78 % (RMSE = 6.9544) and 63.44 % (RMSE = 6.4138) of that of GA (RMSE = 10.1105), respectively. Improvement of the prediction accuracy in the MF model compared to the BUI model suggests that the matrix factorization successfully acquired a high dimensional representation that can characterize the interaction between students and tasks in abacus-based calculations.

Figure 3: Scatter-gram showing the performance prediction by the MF model (Eq. 1), plotted against the true performance scores (Eq. 3). Each dot represents a student-task pair.

6 DISCUSSION

Successful prediction of the performance of each student in a specific task provides the opportunity to give each student the appropriate task necessary for him or her. Implementation of automated system that can give novel tasks with the optimal difficulty level to individual students would realize adaptive learning for a large number of students. Actually, at Digika Co., Ltd., almost all the procedures in teaching the abacus-based calculation are currently

Figure 4: Root mean squared error (RMSE) in the test data for the global average (GA), biased student task (BUI) and matrix factorization (MF) models.

done through the iPad application, SoroTouch; both the presentation of the tasks to the students and the data accumulation process are conducted through the same platform [18]. This kind of educational system may offer the infrastructure for developing and implementing an automated adaptive system.

Further application of more sophisticated algorithms may enhance the prediction accuracy. For example, tensor factorization enables incorporation of the time evolution of each student's learning state, which is omitted in the current study [10, 16]. The current study also omitted the information contained in the data with small sampling sizes. Utilizing different objective function such as divergence between prediction and the data [12] other than simple Frobenius norm of the residuals may lead to a better prediction accuracy using the data with small sample sizes.

7 CONCLUSION

We conducted modeling of student learning status and tasks in abacus-based calculation by utilizing matrix factorization for the student performance data. The matrix consisting of performance scores in student-task pairs is decomposed into two matrices, yielding the distributed representations of each student and each task. Prediction of the students' performances using those decomposed matrices achieved improvement compared to baseline models that used the students' biases and task biases, suggesting that matrix factorization successfully extracted the interaction of the multiple components of each task and each student's learning status in abacus-based calculation.

ACKNOWLEDGMENTS

We wish to thank all students and their parents who support SoroTouch.

REFERENCES

- [1] Barner D., Alvarez G., Sullivan J3., Brooks N., Srinivasan M., Frank M. C. 2016. Learning Mathematics in a Visuospatial Format: A Randomized, Controlled Trial of Mental Abacus Instruction. *87* 4 (2016), 1146–1158.
- [2] Du, F., Yao, Y., Zhang, Q., Chen, F. 2014. Long-term abacus training induces automatic processing of abacus numbers in children. *Perception* 43 (2014), 694–704.
- [3] Lawton W. H. and Sylvestre E. A. 1971. Self Modeling Curve Resolution. *Technometrics* 13, 3 (Aug. 1971), 617–633.

- [4] Hanakawa, T., Honda, M., Okada, T., Fukuyama, H. 2003. Neural correlates underlying mental calculation in abacus experts: a functional magnetic resonance imaging study. *Neuroimage* 19 (2003), 296–307.
- [5] Hatano G., Miyake Y., Binks G. 1977. Performance of expert abacus operators. *Cognition* 5, 1 (1977), 47–55.
- [6] Hatano, G., Osawa, K. . 1983. Digit memory of grand experts in abacus-derived mental calculation. *Cognition* 15, 1–3 (1983), 95–110.
- [7] Hatta, T., and Miyazaki, M. . 1989. Visual imagery processing in Japanese abacus experts. *Imagination, Cognition and Personality* 9, 2 (October 1989), 91–102.
- [8] Hu Y., Geng F., Tao L., Hu N., Du F., Fu K., Chen F. 2011. Enhanced white matter tracts integrity in children with abacus training. *Human Brain Mapping* 32, 1 (January 2011), 10–21.
- [9] T. Kojima. 1954. *The Japanese abacus: its use and theory*. Charles E. Tuttle Company, Inc. of Rutland, Vermont & Tokyo, Japan.
- [10] Kolda, T. G. and Bader, B. W. 2009. Tensor decompositions and applications. *SIAM Rev.* 51, 3 (2009), 455–500.
- [11] Koren, Y., Bell, R., and Volinsky, C. 2009. Matrix factorization techniques for recommender systems. *IEEE Computer Society Press* 42, 8 (2009), 30–37.
- [12] Lee D. D., Seung H.S. 2000. Algorithms for non-negative matrix factorization. In *NIPS'00: Proceedings of the 13th International Conference on Neural Information Processing*, 535–541.
- [13] K. Menninger. 1969. *Number words and number symbols: A cultural history of numbers*. Dover Publications, New York.
- [14] Ministry of Education, Culture, Sports, Science and Technology, Japan. 2008. Shougakko gakushuu shido youryo. Retrieved January 2020 from http://www.mext.go.jp/a_menu/shotou/new-cs/youryou/syo/index.htm
- [15] Stigler, J. W. 1984. "Mental abacus": The effect of abacus training on Chinese children's mental calculation. *Cognitive Psychology* 16 (1984), 145–176.
- [16] Thai-Nghe, N., Drumond, L., Horváth, T., Krohn-Grimberghe, A., Nanopoulos, A., Schmidt-Thieme, L. 2012. Factorization Techniques for Predicting Student Performance. In Santos O, Boticario J (eds) *Educational Recommender Systems and Technologies: Practices and Challenges*. (2012), 129–153. <https://doi.org/10.4018/978-1-61350-489-5.ch006>
- [17] Thai-Nghe, N., Drumond, L., Krohn-Grimberghe, A., and Schmidt-Thieme, L. 2010. Recommender system for predicting student performance. *Proceedings of the 1st Workshop on Recommender Systems for Technology Enhanced Learning (RecSysTEL 2010)* 1, 2 (Aug. 2010), 2811–2819.
- [18] Tokuda K., Hashimoto Y., Yamauchi C., Iwasa K., Yoshida K., Fujiwara N., Sudo A. 2020. Evaluation of Mental Calculation Educational System with Invisible Abacus. In *ICIET 2020: Proceedings of the 2020 8th International Conference on Information and Education Technology (ICIET 2020)*. Association for Computing Machinery, New York, NY, USA. in press.
- [19] Wang, Y., Geng, F., Hu, Y., Du, F., Chen, F. 2013. Numerical processing efficiency improved in experienced mental abacus children. *Cognition* 127, 2 (2013), 149 – 158. <https://doi.org/10.1016/j.cognition.2012.12.004>
- [20] Yao, Y., Du, F., Wang, C., Liu, Y., Weng, J., Chen, F. 2015. Numerical processing efficiency improved in children using mental abacus: ERP evidence utilizing a numerical Stroop task. *Frontiers in Human Neuroscience* 9 (2015), 245. <https://doi.org/10.3389/fnhum.2015.00245>